

Parallel exact diagonalization solver for quantum-electron models

S. N. Iskakov, V. V. Mazurenko

Ural Federal University, Mira St. 19, 620002 Yekaterinburg, Russia

Abstract

We present a parallel computation scheme based on the Arnoldi algorithm for exact diagonalization of quantum-electron models. It contains a selective data transferring method and distributed storage format for efficient computing of the matrix-vector product on distributed computing systems. The performed numerical experiments demonstrated good performance and scalability of our eigenvalue solver. The developed technique has been applied to investigate the electronic properties of Sr_2RuO_4 at experimental temperatures. The role of the spin-flip term in the electronic Hamiltonian was analyzed.

Keywords: DMFT, sparse matrices, matrix-vector product, distributed computing

1. Introduction

Dynamical mean-field theory (DMFT) has been successfully used for studying the electronic properties of a variety of strongly correlated materials in which the strength of the electron-electron interaction energy is comparable to or larger than the kinetic one [1]. Since these characteristics lead to a wealth of physical phenomena such as metal insulator transitions, exotic magnetic structures, and unconventional superconducting phases [2], the DMFT method attracts a great deal of attention. The main underlying idea of this approach is the mapping of the lattice problem onto an effective impurity problem which in turn is solved numerically exactly. For that one can use different numerical techniques such as exact diagonalization (ED), numerical renormalization group (NRG), quantum Monte Carlo method with the Hirsch-Fye algorithm (QMC-HF), or other schemes [1].

Many interesting and promising results were obtained by using QMC methods. For instance, a realistic five-band modeling of the strongly correlated materials is mainly performed by using QMC Hirsch-Fye method. However, there is a number of technical limitations. (i) The simulation temperature of 250 K - 1000 K is much higher than experimental one of 10 K - 100 K. (ii) The Coulomb interaction matrix contains density-density terms only. (iii) The sign problem prevents scientists from calculating the physical properties at sufficiently low temperatures [3]. (iv) The QMC approaches require a maximum entropy method for analytical continuation of the resulting Green's function to real-energy axis.

Email addresses: `iskakoff@ngs.ru` (S. N. Iskakov), `mvv@dpt.ustu.ru` (V. V. Mazurenko)
Preprint submitted to Journal of Computational Physics *April 21, 2010*

Such problems can be solved by using the ED techniques. The full diagonalization method is limited to three-band systems because of the extremely rapid increase of the Hilbert space with the number of the electronic states N_s . For instance, full diagonalization of the Hamiltonian with two impurity orbitals, each coupled to three bath levels ($N_s = 8$), requires diagonalization of matrices with dimension up to 4900 which roughly represents the time and storage limit of what is computationally meaningful.

The diagonalization problem can be simplified by exploiting the extreme sparseness of the Hamiltonian matrices and focusing on the limited number of the states with the lowest energies. It can be done by using Lanczos [4] or Arnoldi [5] approaches. As a result the Hamiltonian corresponding to $N_s = 12$ can be treated at about the same computational cost as the full diagonalization for $N_s = 8$. This improvement allows the application of ED/DMFT to realistic three-band and five-band systems [5]. However, there is memory limitation of the Hamiltonian matrix storage to apply ED/DMFT to more complicate physical systems. To date the biggest matrix that can be diagonalized by this approach on a modern workstation is about $6435^2 \times 6435^2$ which corresponds to $N_s = 15$ [6].

In this paper we present a sparse matrix-vector product (SpMV) parallelization strategy for effective solving ED/DMFT equations on distributed computing systems. For these purposes we suggest two improvements of the original ED/DMFT method [1]: a distributed storage format for sparse large matrix and a selective data transferring. The former allows the application of the ED/DMFT method for system with the number of electronic states more than $N_s = 15$. The selective data transferring method substantially reduces amount of data transmitted between the processors of a distributed computing system, which improves SpMV performance.

The paper organized as follows. In section 2, we presents the basic steps of the DMFT theory. In section 3, the developed eigenvalue solver and its parallel performance are given. We also give approbation of our computational scheme for a one-band Hubbard model on square lattice. For purpose of illustration in section 4 we will use a three-band model for Sr_2RuO_4 compound.

2. Theory

One way to describe many-particle phenomena such as high-temperature superconductivity, heavy fermions and others is to solve the Hubbard Hamiltonian [7]

$$H = - \sum_{i,j,\sigma} t_{ij} c_{i\sigma}^+ c_{j\sigma} + U \sum_i n_{i\uparrow} n_{i\downarrow} \quad (1)$$

where $c_{i\sigma}^+$ ($c_{i\sigma}$) is creation (annihilation) operators of fermion, t_{ij} is a hopping integral, U is the on-site local Coulomb interaction, n_i is the particle number operator. It is a very complicated problem due to the exponential growth of the Hamiltonian and, hence, the direct solution of the Hubbard Hamiltonian is only possible for small model clusters [8].

In case of the real systems one can use the Dynamical Mean-Field Theory [1] in which the lattice problem is reduced to a single-site effective problem. Thus, such a mean-field consideration leads to freezing spatial correlations. Instead of the Hubbard Hamiltonian

we solve the Anderson impurity model which is given by

$$H_{AIM} = \sum_{k,\sigma} V_{kd\sigma} [d_\sigma^+ c_{k\sigma} + c_{k\sigma}^+ d_\sigma] + \sum_{k,\sigma} \varepsilon_{k\sigma} c_{k\sigma}^+ c_{k\sigma} + \varepsilon_d (n_\uparrow + n_\downarrow) + U n_\uparrow n_\downarrow, \quad (2)$$

where $N_k = N_s - 1$ is a number of the bath states, $d_\sigma^+(d_\sigma)$ and $c_{k\sigma}^+$ ($c_{k\sigma}$) are creation (annihilation) operators for fermions with spin σ associated with the impurity site and with the state k of the effective bath, respectively, ε_k and ε_d are the energy of bath and impurity, V_{kd} is a hybridisation integral of impurity and bath states. The main task of the ED/DMFT is to diagonalize the impurity Hamiltonian H_{AIM} to compute the impurity Green function which can be written as

$$G(i\omega_n) = \frac{1}{Z} \sum_{\nu\mu} \frac{\|\langle \mu | c_\sigma^+ | \nu \rangle\|^2}{E_\nu - E_\mu - i\omega_n} [e^{-\beta E_\nu} + e^{-\beta E_\mu}], \quad (3)$$

where $|\nu\rangle$ ($|\mu\rangle$) and E_ν (E_μ) are eigenvectors and eigenvalues of H_{AIM} and $Z = \sum_\mu \exp(-\beta E_\mu)$ is the partition function. The exponential factor in (3) indicates that at large enough β only a small number of the lowest eigenstates needs to be calculated [4]. In this paper, we develop a high performance parallel technique for solving the eigenvalue problem of the Anderson impurity model Hamiltonian matrix (2) on distributed computing systems.

3. Distributed eigenvalue solver

A few iterative numerical algorithms such as a power method, a Lanczos method, an implicitly restarted Arnoldi method [9], the conjugate gradient (CG) method [10] have been proposed to solve the eigenvalue problem for large sparse matrices. Most of them are based on iterative multiplication of the Hamiltonian matrix H by the trial vector ν . Thus they are ideally suited for parallelization.

An efficient implementation of the Lanczos method for symmetric multiprocessing computers with large shared memory was proposed in Ref.[11] for solving the Heisenberg model. The authors have demonstrated a perfect scalability of their computational scheme. It was possible since they used a SMP machine with eight cores which were sitting on the same board. As we will show below it is not the case for a distributed architecture.

Recently, the ED results for a five-band DMFT problem were reported by Liebsch [6]. Judging by a short description of the technical details they have also used a SMP machine with large shared memory and 32 processors on the board. The aim of our work is to modify the exact diagonalization method for solving DMFT equations on distributed computing systems.

3.1. Distributed CSR format and selective data transferring

Since the Hamiltonian matrix H is extremely sparse the performance of the $H \times \nu$ operation strongly depends on the storage format and SpMV algorithm we use. The memory subsystem and, more specifically, the memory bandwidth is identified as the main performance bottleneck of the SpMV routine [12, 13]. This performance bottleneck is due to the fact that SpMV performs $O(nnz)$ operations where nnz is a number of the

nonzero elements of the sparse matrix, which means that most of the data are accessed in a streaming manner and there is little temporal locality. To avoid indirect memory access some sparse matrix storage format is used. One of the most widely used storage format for sparse matrices is Compressed Sparse Row (CSR) format [14, 15] which stores all the non-zero elements of the Hamiltonian in contiguous memory locations (H_{values} array) and uses two additional arrays for indexing information: row_ind contains the start of each row within the non-zero elements array and col_ind contains the column number associated with each non-zero element. The size of the H_{values} and col_ind arrays are

$$H = \begin{pmatrix} h_{00} & 0 & 0 & 0 & h_{04} & 0 \\ 0 & h_{11} & 0 & 0 & 0 & h_{15} \\ 0 & 0 & h_{22} & 0 & 0 & 0 \\ 0 & 0 & 0 & h_{33} & 0 & 0 \\ h_{40} & 0 & 0 & 0 & h_{44} & 0 \\ 0 & h_{51} & 0 & 0 & 0 & h_{55} \end{pmatrix}$$

↓

$$row_ind = (0 \ 2 \ 4 \ 5 \ 6 \ 8 \ 10)$$

$$col_ind = (0 \ 4 \ 1 \ 5 \ 2 \ 3 \ 0 \ 4 \ 1 \ 5)$$

$$H_{value} = (h_{00} \ h_{04} \ h_{11} \ h_{15} \ h_{22} \ h_{33} \ h_{40} \ h_{44} \ h_{51} \ h_{55})$$

Figure 1: Example of CSR storage format. The original matrix H is compressed into three arrays row_ind , col_ind and H_{value} .

equal to the number of the non-zero elements (nnz), while the size of the row_ind array is equal to the number of rows ($nrows$) plus one. An example of the CSR format for a sparse 6×6 matrix is presented in Figure 1. In order to organize the matrix-vector multiplication operation one can use the following listing (Listing 1).

Listing 1: Sparse Matrix-Vector Multiplication using CSR format

```

1  for ( i=0; i<Hdim; i++)
2      for ( j=row_ind[ i ] ; j<row_ind[ i +1]; j++)
3          y[ i ] += H[ j ] * v[ col_ind[ j ] ];

```

In the framework of the standard SpMV procedure with the standard CSR format all three arrays and vector should be stored on the same board. It is impossible when the size of the Hamiltonian is larger than the memory of a single machine. Therefore the first goal of our work is to design method for arrays distribution over CPU mesh to overcome the problem of large matrix storage.

We use the standard CSR storage format as a starting point. The compressed Hamiltonian as well as the accompanied arrays are distributed over processors grid. It reduces required memory space of a single CPU (table 1). The Hamiltonian matrix is stored distributively all the time. For this purpose we have developed the library written in C++ for initialization and working with different elements of distributed arrays.

The second goal of our work is to organize effective and optimal communications between processors to reduce amount of transmitted data. To solve this problem we have developed communication procedures for operating elements of the matrix stored distributively. The subroutines can be divided into two classes. The first one is used for

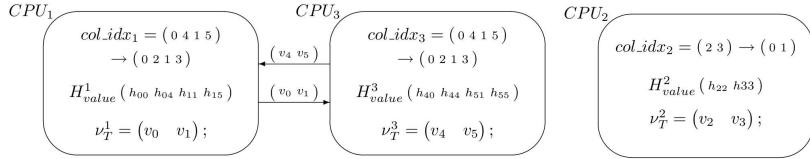


Figure 2: Example of Distributed CSR storage format running on 3 CPU. The arrows denote the communications between CPUs.

exchanging data between CPUs based on one-sided communication functions of MPI. For optimal performance and memory usage we used MPI_Get function, because in case of exchanging large amount of data MPI_Put subroutine needs significantly more memory. The second class of the subroutines is directed to determine groups of interacting CPUs. In a sense the main advantage of the distributed CRS format we proposed is that each CPU is to communicate with only a few CPUs to receive the different parts of the Arnoldi vector during simulation. For example, while solving eigenvalue problem for the matrix with the size of 165,636,900 only about 30 of a 256 CPUs are needed for each CPU to receive required parts of the vector.

The figure 2 shows an example of distributed CSR format for 6×6 matrix in case of three processors. One can see that the original compressed matrix (Fig.1) and the array col_idx is distributed into 3 arrays. Since the matrix-vector multiplication each CPU needs only a small part of the whole vector then the index array can be considerably reduced. Thus the approach we proposed naturally leads to reduction of single CPU memory requirement for the trial vector storage. CPU 1 and CPU 3 are to transmit the different parts of the vector to each other.

3.2. Performance

To test the performance of the developed technique we have performed the ED/DMFT calculations for a one-band Hubbard model on a square lattice with the on-site Coulomb interaction $U = 2$ eV and the nearest hopping integral $t = 0.5$ eV (Fig.3). The calculations were carried out for $N_s=16$. The resulting Green function is presented in Fig. 3. One can see that there is the famous three peaks structure (quasiparticle peak, lower and upper Hubbard bands) that agrees with results of the previous theoretical investigations [1].

Table 1 shows the performance of the modified ED/DMFT calculation scheme on $N_p = 32, 64, 128, 256$, and 512 processors. These performance measurements were made as follows. The total elapsed time and time of communications between processors were measured by MPI_Wtime function. The required memory was measured by the tools installed on a particular parallel cluster system.

Fig.4 shows the scaling of the computation time for 1 ED/DMFT iteration with a maximum vector length of 11,778,624. One can see that the speedup of our calculation scheme is far from the ideal scaling. The main reason for that is time required for communication between different CPUs. Another operation of our computational scheme which takes much time is the construction of the Hamiltonian matrix (Table 1).

We can also compare the speedup of our algorithm in case of the different sizes of the Hamiltonian matrix. For $N_s=16$ the ratio $T_{256}/T_{512} = 1.58$ is larger and closer to

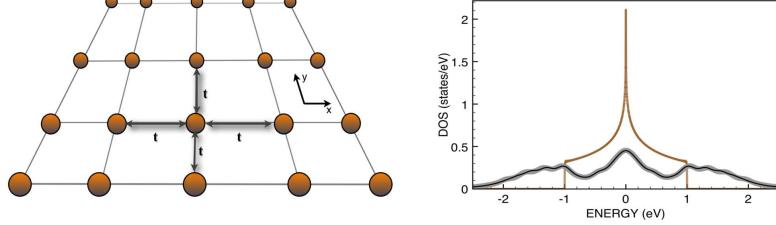


Figure 3: (left) Schematic representation of the square lattice with the nearest neighbors couplings. (right) Spectral functions obtained from ED/DMFT calculations with $N_s = 16$ (gray bold line). The dashed brown line corresponds to the non-interacting density of states.

the ideal one than $T_{32}/T_{64} = 1.26$ in case of $N_s = 14$. Based on the obtained performance data we can estimate the number of processors we need for $N_s = 18$ and $N_s = 20$. For instance, in case of $N_s = 18$, for the largest sector one needs to store about the 82 billions of non-zero elements. It requires about 1.5 Tb of memory and can be solved by using 2000 cores cluster. Investigation of the systems with $N_s = 20$ leads to operating about 1194 billions of the non-zero elements and it is impossible to solve this problem using our scheme.

It is interesting to compare the performance of the developed technique with that reported in Ref. [16]. For $N_s = 12$ the authors of the paper have estimated that the time of the one iteration with 300 lowest eigenstates was less than 30 min if 16 processors were employed in parallel. In our case the similar calculations take us about 45-50 min. Such a time difference can be explained by taking into account that the author used a workstation with shared memory and processors on the same board.

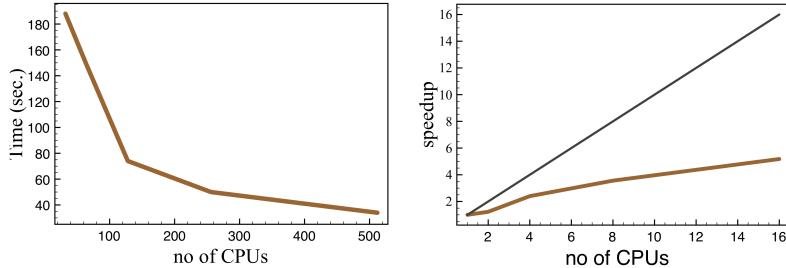


Figure 4: Scaling of the total computation time for 1 ED/DMFT iteration with number of used CPUs. The number of CPU in the speedup dependence (right) is normalized by 32.

It is important to discuss the possible ways to improve our computation scheme. Firstly, The results of the previous work [17] have demonstrated that it would be better not to store the Heisenberg matrix, but to evaluate the matrix elements whenever needed. Secondly, since we divide the full Hamiltonian matrix into sub-matrices of equal number of the elements there is an irregular distribution of the non-zero elements. It results in unequal partition of workloads among processors. This load-imbalance problem can

Table 1: Performance of the developed eigenvalue solver. N_p is a number of processors and nnz_{max} is a maximum number of the non-zero elements per row, M_{exch} and M_{data} are amount of transferred data per processor and memory requirement per processor (in MB). T_{total} , T_{comm} and T_{setup} are the total computational, communication time and time for generating Hamiltonian matrix (in sec.), respectively.

N_s	H_{dim}	nnz_{max}	N_p	M_{exch}	M_{data}	T_{setup}	T_{comm}	T_{total}
14	11,778,624	13	32	23	300	17	66	188
14	11,778,624	13	64	22	150	10	71	149
14	11,778,624	13	128	15	75	3.2	47	74
14	11,778,624	13	256	7	40	1.7	35	50
14	11,778,624	13	512	4	20	1.0	22	34
16	165,636,900	15	256	120	500	52	330	602
16	165,636,900	15	512	54	270	25	200	382

be solved by partitioning the Hamiltonian matrix in a computational space where the optimal number of the non-zero elements is defined to minimize the load-balance and communication costs.

4. Numerical results

Using the developed exact diagonalization technique we have investigated the electronic structure of Sr_2RuO_4 compound. This system is of interest due to a number of unconventional physical properties. For instance, Sr_2RuO_4 demonstrates an unconventional superconductivity at temperatures below 1 K. Moreover, photoemission and optical experiments have shown substantial mass renormalization effects. In the previous theoretical investigations the electronic structure of Sr_2RuO_4 was studied within LDA+DMFT method using the QMC-HF as the impurity solver. For instance, Liebsch and Lichtenstein [18] have performed multiband DMFT calculations to explain the discrepancy between photoemission and de Haas-van Alphen data. It was found that there is a charge transfer from $3d_{xz,yz}$ states to $3d_{xy}$ states. The similar QMC-HF method was used by the authors of the paper [19] to describe some features of photoemission spectra.

In previous investigations the lowest simulation temperature was 15 meV, which is larger those at which experiments were performed. The imaginary time QMC data were analytically continued by the maximum entropy method. Since the QMC-HF/DMFT method operates only with density-density interaction terms the spin-flip term in the impurity Hamiltonian was not taken into account. At the same time the effect of such an interaction on electron spectra was mentioned in a number of works [18].

To take into account the spin-flip effects in our ED/DMFT calculation we have diagonalized the following impurity Hamiltonian:

$$H = \sum_{m\sigma} (\epsilon_m - \mu) n_{m\sigma} + \sum_{k\sigma} \epsilon_k n_{k\sigma} + \sum_{mk\sigma} V_{mk} [d_{m\sigma}^+ c_{k\sigma} + H.c.] + \sum_m U n_{m\uparrow} n_{m\downarrow} \\ + \sum_{m < m' \sigma \sigma'} (U' - J\delta_{\sigma\sigma'}) n_{m\sigma} n_{m'\sigma'} - \sum_{m \neq m'} J' [d_{m\uparrow}^+ d_{m\downarrow} d_{m'\downarrow}^+ d_{m'\uparrow} + d_{m\uparrow}^+ d_{m\downarrow}^+ d_{m'\uparrow} d_{m'\downarrow}]. \quad (4)$$

Here $d_{m\sigma}^{(+)}$ are annihilation (creation) operators for impurity electrons on orbital m with spin σ , $U(U')$ is the intra-orbital (inter-orbital) impurity Coulomb energy, J is an intra-atomic exchange integral and J' is the spin-flip interaction. The account of the latter interaction is possible in the framework of the exact diagonalization and recently developed CT-QMC calculation scheme. In accordance with Ref.[18] these parameters were chosen to be $U = U' = 1.2$ eV, $J = 0.2$ eV and $J' = 0.1$ eV. The simulation temperature was taken to be $T = 200$ K that is close to the experimental values used in Ref.[18]. Thus for a total of about 20 excited states the Green's functions are calculated via the Lanczos procedure. Of course, this number increases at higher temperatures, and if Boltzmann factors smaller than 10^5 are included for higher precision.

The resulting Green functions obtained for $N_s = 15$ are presented in Fig.5. The spectral functions without the spin-flip term agree well with those presented in Ref. [18]. One can see that the account of the spin-flip term leads to an additional renormalization of the spectrum near the Fermi level. It means that other non-diagonal elements of the

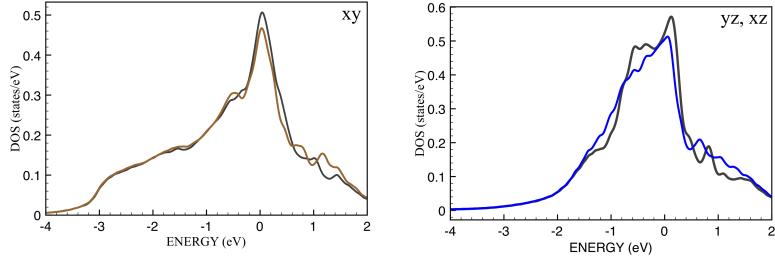


Figure 5: Densities of states obtained from DMFT calculations for $T = 200$ K with (brown and blue lines) and without (gray lines) the spin-flip term.

full U-matrix can play an important role for correct description of the experimentally observed spectra. The approach we used here is also well suited for analyzing the ground and few excited states. Such an analysis can help us to study the experimentally observed superconducting phase.

5. Conclusions

To conclude, we developed a distributed storage format and a selective data transferring procedure for distributed sparse matrix-vector multiplication. They were implemented in ED solver for dynamic mean field theory problem and showed good speedup and scalability. The obtained numerical results look quite promising. The developed approach can be used for solving Heisenberg model on computer clusters with distributed memory.

6. ACKNOWLEDGMENTS

We would like to thank A.O. Shorikov for his technical assistance with test calculations. This work is supported by the scientific program “Development of scientific potential of universities” N 2.1.1/779, by the scientific program of the Russian Federal

Agency of Science and Innovation N 02.740.11.0217, the grant program of President of Russian Federation MK-1162.2009.2., RFFI 10-02-00546. The calculations have been performed on the cluster of University Center for Parallel Computing of Ural Federal University and the Brutus cluster at ETH Zurich.

- [1] A. Georges, G. Kotliar, W. Krauth, M. J. Rozenberg, Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions, *Rev. Mod. Phys.* 68 (1) (1996) 13.
- [2] M. Imada, A. Fujimori, Y. Tokura, Metal-insulator transitions, *Rev. Mod. Phys.* 70 (4) (1998) 1039–1263.
- [3] E. Gorelov, T. O. Wehling, A. N. Rubtsov, M. I. Katsnelson, A. I. Lichtenstein, Relevance of the complete Coulomb interaction matrix for the Kondo problem: Co impurities in Cu hosts, *Phys. Rev. B* 80 (15) (2009) 155132.
- [4] M. Capone, L. de' Medici, A. Georges, Solving the dynamical mean-field theory at very low temperatures using the Lanczos exact diagonalization, *Physical Review B (Condensed Matter and Materials Physics)* 76 (24) 245116.
- [5] C. A. Perroni, H. Ishida, A. Liebsch, Exact diagonalization dynamical mean-field theory for multi-band materials: Effect of Coulomb correlations on the Fermi surface of $Na_{0.3}CoO_2$, *Physical Review B (Condensed Matter and Materials Physics)* 75 (4) 045125.
- [6] H. Ishida, A. Liebsch, Fermi-liquid, non-Fermi-liquid, and Mott phases in iron pnictides and cuprates, *Phys. Rev. B* 81 (5) (2010) 054513.
- [7] J. Hubbard, Electron Correlations in Narrow Energy Bands, *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences* 276 (1365) (1963) 238–257.
- [8] S. Yamada, T. Imamura, M. Machida, 16.447 TFlops and 159-Billion-dimensional Exact-diagonalization for Trapped Fermion-Hubbard Model on the Earth Simulator, in: SC '05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing, IEEE Computer Society, Washington, DC, USA, ISBN 1-59593-061-2, 44, 2005.
- [9] K. Maschhoff, D. Sorensen, A portable implementation of ARPACK for distributed memory parallel architectures, URL citeseer.ist.psu.edu/maschhoff96portable.html, 1996.
- [10] M. R. Hestenes, E. Stiefel, Methods of Conjugate Gradients for Solving Linear Systems, *Journal of Research of the National Bureau of Standards* 49 (6) (1952) 409–436.
- [11] J. Schnack, P. Hage, H.-J. Schmidt, Efficient implementation of the Lanczos method for magnetic systems, *J. Comput. Phys.* 227 (9) (2008) 4512–4517, ISSN 0021-9991.
- [12] K. Kourtis, G. Goumas, N. Koziris, Optimizing sparse matrix-vector multiplication using index and value compression, in: CF '08: Proceedings of the 5th conference on Computing frontiers, ACM, New York, NY, USA, ISBN 978-1-60558-077-7, 87–96, 2008.
- [13] W. K. Anderson, W. D. Gropp, D. K. Kaushik, D. E. Keyes, B. F. Smith, Achieving high sustained performance in an unstructured mesh CFD application, in: Supercomputing '99: Proceedings of the 1999 ACM/IEEE conference on Supercomputing (CDROM), ACM, New York, NY, USA, ISBN 1-58113-091-0, 69, 1999.
- [14] J. Mellor-crummey, J. Garvin, Optimizing Sparse Matrix-Vector Product Computations Using Unroll and Jam, *International Journal of High Performance Computing Applications* 18 (2003) 2004.
- [15] E.-J. Im, Optimizing the Performance of Sparse Matrix-Vector Multiplication, Ph.D. thesis, EECS Department, University of California, Berkeley, 2000.
- [16] A. Liebsch, N.-H. Tong, Finite-temperature exact diagonalization cluster dynamical mean-field study of the two-dimensional Hubbard model: Pseudogap, non-Fermi-liquid behavior, and particle-hole asymmetry, *Phys. Rev. B* 80 (16) (2009) 165126.
- [17] J. Schnack, P. Hage, H.-J. Schmidt, Efficient implementation of the Lanczos method for magnetic systems, *Journal of Computational Physics* 227 (9) (2008) 4512 – 4517, ISSN 0021-9991.
- [18] A. Liebsch, A. Lichtenstein, Photoemission Quasiparticle Spectra of Sr_2RuO_4 , *Phys. Rev. Lett.* 84 (7) (2000) 1591–1594.
- [19] Z. V. Pchelkina, I. A. Nekrasov, T. Pruschke, A. Sekiyama, S. Suga, V. I. Anisimov, D. Vollhardt, Evidence for strong electronic correlations in the spectra of Sr_2RuO_4 , *Phys. Rev. B* 75 (3) (2007) 035122.